

Regrid Pack Documentation

INTRODUCTION

This module provides access through Python to the collection of Fortran programs in REGRIDPACK, which is a collection of programs produced at the National Center for Atmospheric Research by John C. Adams for linear or cubic interpolation in one, two, three or four dimensions.

RESTRICTIONS

This module provides regridding by interpolation only. In other words, the output coordinate vector can not extend beyond the limits of the input coordinate vector.

Each coordinate vector must be monotonically increasing or decreasing.

Missing data is not allowed in the input data.

CAPABILITIES

This module allows linear or cubic interpolation in one, two, three or four dimensions. The computation can use the grid vectors associated with the input and the output data or just their sizes if the input and output vectors are uniformly spaced. However, the selection of the uniform option must apply to all the dimensions in the requested interpolation. In addition, the computation in the nonuniform case can use the log of the grid vectors chosen dimension by dimension. It is possible to regrid a subset of the dimensions in the input data. A utility function is provided to generate a gaussian or a uniform grids to use as a output coordinate vector.

ORGANIZATION

This module is object oriented for simplicity. It is organized as a single class called Regrid, which begins with a capital letter. Python is case sensitive. It contains the single function, `rgrd`, which performs the computation in one, two, three or four dimensions.

Access to the function `rgrd` is provided through a simple two step process. The first step is making an instance of the class passing the coordinate vectors and associated information. The second step is calling the regridding function with the data in the argument list and an optional missing data value to request a check for the presence of a missing data.

HELP

To obtain a prescription for making an instance, type

```
adamsregrid.help('Regrid')
```

To acquire instructions on the use of the `rgrd` function, type

```
adamsregrid.help('rgrd')
```

To look at a general one dimensional example, type

```
adamsregrid.help('OneDexample')
```

To look at a general four dimensional example, type

```
adamsregrid.help('FourDexample')
```

DOCUMENTATION

Documentation written to the file regridmodule.doc can be produced after importing the adamsregrid module by typing

```
adamsregrid.document()
```

As an alternate to using the help package, online documentation is available from three individual doctstrings by using

```
print adamsregrid.Regrid.__doc__ -- documentation for the module.  
print adamsregrid.Regrid.__init__.__doc__ -- documentation for making an instance.  
print adamsregrid.Regrid.rgrd.__doc__ -- documentation for the rgrd method.
```

TESTING

After importing adamstest, typing

```
cdat adamstest.py
```

generates some testing of the adamsregridmodule using analytical functions as fields. It also writes a hard copy of the documentation to the file regridmodule.doc and a copy of the information displayed on the screen to screen.asc.